

REMARKS/ARGUMENTS

Claims 1-24 are present for examination:

- Claims 1-2, 7, 13-14, and 20-24 stand rejected under 35 USC 103(a) as unpatentable over the admitted prior art (APA) in view of Smyers and further in view of Lau.
- Claims 3-6, 8, 9, 10-12, and 15-19 stand rejected under 35 USC 103(a) as unpatentable over the admitted prior art (APA) in view of Smyers, further in view of Lau as applied to claim 1, and further in view of Messer.

Examiner Truong is thanked for the Office Action dated September 25, 2006, and especially for the comments contained in the Response to Argument section starting at page 7 of the Office Action. While Applicant respectfully traverses the conclusions set forth therein, the further explanation of the Examiner's reasoning has helped clarify the issues in the present application.

Independent Claims 1, 7, 13, and 20

Independent claim 13 is directed to a method of displaying information on a monitor screen of a processing system connected to a digital audio playback device (DAPD). The method requires, *inter alia*:

executing a reverse DAPD application programming interface (API), wherein the step of executing the reverse DAPD API enables the digital audio playback device to *access and control* [a] user interface associated with an application program [of the connected processing system] and displayed on a monitor screen associated with the connected processing system.

(emphasis added). Independent claims 1, 7, and 20 contain similar limitations.

In rejecting the present claims, page 7 of the Office Action cites to certain isochronous data transfers performed by Smyers system, citing the example of a video recorder coupled to a video monitor. The Office Action concludes:

Smyers teaches the video cassette recorder implements the API to transfer the data of the video cassette recorder for displaying. Since, the data from the recorder is displayed on the video monitor, the user interface of the video monitor is accessed and controlled by the video cassette [recorder] by using the API to transfer the data of the video cassette [recorder] for display. The data of [the] video cassette [recorder] cannot be displayed on the video monitor without accessing the user interface of the video monitor.

Applicant respectfully submits that this conclusion is misplaced. Applying the cited example to the present claims,¹ it is respectfully submitted that that Smyers fails to disclose or suggest a DAPD API which:

- enables the video recorder to access and control the user interface associated with an application program of the video monitor;
- enables the video recorder to access and control the user interface displayed on a monitor screen associated with video monitor.

The isochronous data transfers relied upon by the Office Action are described more fully at Smyers col. 5:59 - col. 9:28 and in relation to Figure 2. Again with respect to the example cited by the Office Action, Smyers teaches that its API 20 performs two types of data transfers.

The first type of data transfer will be referred to, for the purposes of the present discussion, as a “transfer out.” With particular reference to Smyers col. 7:4-14, an application program 22 of the video recorder fills the recorders data buffers 32, 34, 36 with data to be transferred to the video monitor. The video recorder API 20 then manages, in cooperation with a physical bus interface 26, the transfer of the data from the appropriate buffer to the data bus 28.

The second type of data transfer will be referred to, for the purposes of the present discussion, as a “transfer in.” With particular reference to Smyers col. 7:14-23, an API 20 of the video monitor manages the transfer of data from the data bus 28 to the video

¹ It will be understood that the example is cited for explanatory purposes only and should not be construed as limiting the claims to a video recorder and an associated video monitor.

monitor data buffers 32, 34, 36. An application program 22 then reads the data from the appropriate buffer 32, 34, 36 during an appropriate time period.

Thus, in the system of Smyers, an application program of the video recorder stores video data to the video recorder's data buffers. An API of the video recorder manages the transfer of the video data from the buffers to the data bus (a transfer out). An API of the video monitor manages the transfer the video data from the data bus to the data buffers of the video monitor (a transfer in). An application program of the video monitor then reads the data from the monitor's buffers.

Applicant specifically traverses the contention that, since data from the recorder is displayed on the video monitor, the user interface of the video monitor is accessed and controlled by the video recorder by using the API to transfer video data for display. First, the video recorder API merely manages the transfer of data from the video recorder to the data bus (a transfer out). Moreover, Smyers teaches only that the video monitor application program reads the data at the appropriate time. The manner in which the video monitor application program displays or otherwise utilizes the video data (if at all) is not specified by Smyers. Indeed, Smyers fairly teaches that any such use or display of the video data is within the purview of the video monitor application program.

Thus, Smyers fairly teaches that the video recorder includes an API which is used to manage the *transfer of data to the data bus* (i.e., to perform a transfer out). Applicant submits that Smyers does not fairly teach or suggest that the video recorder contains a reverse DAPD API which enables the video recorder to *access and control the user interface* associated with an application program of the video monitor.

Applicant also specifically traverses the contention that the data of the video recorder cannot be displayed without accessing the user interface of the video monitor. First, and as noted above, the video recorder API simply manages the transfer of video data from the video recorder to a data bus (again, a transfer out). Upon being presented with the video data, the video monitor API manages the transfer of the data to the video

monitor's data buffers (*i.e.*, to performs a transfer in), whereupon the data is used as needed by the video monitor application program.

Thus, it cannot fairly be said that the video recorder API must access the user interface of the video monitor in order to display the video data. Again, and as noted above, the video recorder API merely *manages the transfer of the data to the data bus*, with the receipt and use (if any) of the data being managed and controlled by the video monitor's API and application program. In contrast, the present claims require that the video monitor API *access and control the user interface* associated with the video monitor application program and displayed the video monitor.² Smyers does not fairly teach or suggest this requirement.

As the references applied in the Office Action fail to teach or suggest all of the limitations of the present claims, it is submitted that the Office Action fails to establish a *prima facie* case of obviousness with respect to all pending claims 1-24.³

Dependent claims 2, 8, 14, and 21

These claims require that the reverse DAPD API include executable instructions which communicate with and control the operation of the user interface application program.

As discussed more fully above, and again continuing with the example of a video recorder and monitor, Smyers teaches a system in which the video monitor API manages the transfer of data to the data bus. It cannot fairly be said that the video monitor includes a reverse DAPD API which controls the operation of the video monitor's application program.

² Again, it will be understood that the video recorder and monitor are presented for explanatory purposes only and should not be construed as limiting the claims.

³ MPEP 2143.

Dependent claims 3-6, 8, 9, 10-12, and 15-19

The Office Action acknowledges that the APA, Smyers, and Lau do not teach an API which identifies a manufacturer of the DAPD, and that the reverse DAPD is capable of causing in identity of the manufacturer to be displayed. The Office Action cites to Messer col. 11:59-64 to remedy this deficiency. The Office Action further asserts that Messer teaches an API which identifies a manufacturer of the digital audio playback device.

This contention is respectfully traversed. Following the example presented by the Office Action, Messer fairly teaches that the manufacturer of the *video monitor* is identified and that a video window is generated according to the capabilities of the *video monitor*. In contrast, the present dependent claims are directed to the manufacturer of the digital audio playback device – in other words, a device which would be connected to the video monitor.

Consequently, it is submitted that Messer fails to remedy the acknowledged deficiency in the prior art and that the Office Action fails to establish a *prima facie* case of obviousness with respect to dependent claims 3-6, 8, 9, 10-12, and 15-19.

Other Claims

It is submitted that dependent claims not specifically addressed above are directed to patentable subject matter at least by virtue of their dependency from their respective base claims.

CONCLUSION

For at least the foregoing reasons, it is submitted that claims 1-24 distinguish patentably and non-obviously over the prior art of record. An early indication of allowability is earnestly solicited.

Respectfully submitted,

DRIGGS, HOGG & FRY CO., L.P.A.



John J. Fry Reg. No. 35,873
Driggs, Hogg & Fry Co., L.P.A.
38500 Chardon Road
Willoughby Hills, Ohio 44094
Phone: 1.440.391.5100
Fax: 1.440.391.5101

Please Mail Correspondence to:
Yan Glickberg, Esq.
Philips Intellectual Property & Standards
P.O. Box 3001
Briarcliff Manor, NY 10510-8001
Phone: +1 914 333-9618